UCRL- PROC-205443

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Multi Dimensional Phase Only Filter

K.S. Gudmundsson, A. A. S. Awwal*
** Department of Electrical Engineering, Wright State University, Dayton, OH. 45435*

August 2, 2004

SPIE Annual Meeting 2004 – The International Symposium on Optical Science and Technology
Denver, Colorado
August 2 -5, 2004

Multi Dimensional Phase Only Filter

Karl S. Gudmundsson* and Abdul A.S. Awwal**
* Department of Electrical Engineering, Wright State University, Dayton, OH. 45435
** University of California, Lawrence Livermore National Laboratory, Livermore, CA. 94551
E-mail: kgudmund@cs.wright.edu

ABSTRACT

Today's sensor networks provide a wide variety of application domain for high-speed pattern classification systems. Such high-speed systems can be achieved by the use of optical implementation of specialized POF correlator. In this research we discuss the modeling and simulation of the phase only filter (POF) in the task of pattern classification of multi-dimensional data.

Keywords: phase only filter, multi-sensor data classification, eNose, pattern recognition

1 INTRODUCTION

A system composed of multiple sensors for data acquisition requires the analysis of multiple data signals in order to classify the input data. One such system is the electronic Nose system (eNose). An electronic nose (e-nose) is an instrument that combines gas sensor arrays and pattern analysis techniques, for the detection, identification or quantification of volatile compounds[1]. The multivariate response of an array of chemical gas sensors with broad and partially overlapping selectivity can be utilized as an "electronic fingerprint" to characterize a wide range of odors or volatile compound by utilizing pattern-recognition. In this paper, data from such a system is analyzed and classified using the POF. This data is, in fact, 1D data from multiple data sources, thus multi-dimensional data. In 1961 Bellman[2] introduced the concept of the "curse of dimensionality" which refers to the exponential growth of hypervolume as a function of dimensionality. Therefore, in order to analyze and classify multidimensional data using conventional methods such as Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), or the K Nearest Neighbors (KNN) requires huge feature vectors or dimension reduction. In all cases the test sample is compared to the trained database one by one requiring much computational power.

However, application wise, the eNose is just one such possible application area. In this day and age, urban biological defense is a reality. Environmental monitoring, medical monitoring, and public health surveillance are a few of the key components to be considered in such a situation. Environmental detectors that can be used in urban environments for early detection of biological aerosols is one application requiring a high speed correlation technique. Current systems are not up to date and will cause great casualties. They rely on biological point collectors or point detectors. Similarly, applications in bomb identification for airport security might use multiple sensors sensitive to different chemicals, or other quantifiable signal generating material.

2 ENOSE DATA

In the following example, the eNose data is collected using four Figaro gas sensors. Each sensor is sensitive to different gases, such as air contaminants and solvent vapors. When using an array of four sensors, a wide variety of odors can be analyzed and classified. For example, odors of 3 different coffee types can easily be classified using a relatively small database. The following, figure 1, shows one such experiment. In this experiment, a database of 15 samples, 5 samples from each of the 3 different classes of coffee were trained and used to identify a test sample using PCA, LDA, and KNN classifier.

In the PCA method our objective is to perform dimensionality reduction while preserving as much af the randomness in the high dimensional space as possible. There are basically two approaches to dimensionality reduction. Those are the feature extraction and the feature selection methods. In the problem of feature extraction, given a feature space:

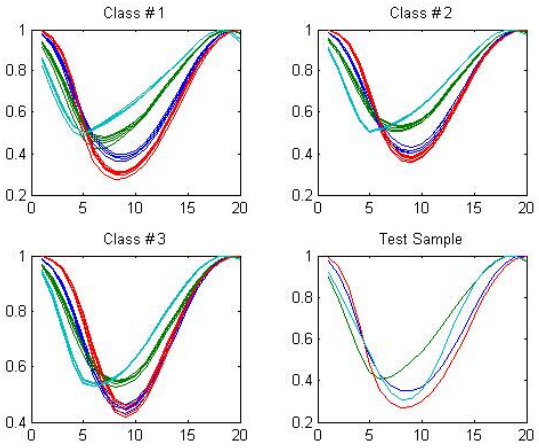

Figure 1. Three classes of patterns and one test pattern

$$x_i \in R^N \tag{1}$$

we want to find a mapping:

$$y = f(x): R^N \rightarrow R^M \; whith \, M < N \tag{2}$$

such that the transformed feature vector:

$$y_i \in R^M \tag{3}$$

preserves most of the information or structure in R^N. An optimal mapping is one that will result in no increase in the minimum probability of error. In general an optimal mapping will be a non linear function. However, there is no systematic way to generate non linear transforms. Therefore, feature extraction is commonly limited to linear transforms such as:

$$\begin{bmatrix} x_1 \\ x_2 \\ : \\ x_N \end{bmatrix} \rightarrow \begin{bmatrix} y_1 \\ y_2 \\ : \\ y_M \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & .. & .. & w_{1N} \\ w_{21} & w_{22} & .. & .. & w_{2N} \\ : & : & . & & : \\ : & : & & . & : \\ w_{M1} & w_{M2} & .. & .. & w_{MN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ : \\ x_N \end{bmatrix} \tag{4}$$

If we let ***x*** be represented as a linear combination of orthonormal basis vectors:

$$x = \sum_{i=1}^{N} y_i \varphi_i \; where \, \varphi_i \left| \varphi_j \right. = \begin{cases} 0 \;\; i \neq j \\ 1 \; i = j \end{cases} \tag{5}$$

We can represent ***x*** with only M (M < N) of the basis vectors by replacing the components $[y_{M+1},\ldots,y_N]^T$ with some preselected constants b_i.

$$x'(M) = \sum_{i=1}^{M} y_i \varphi_i + \sum_{i=M+1}^{N} b_i \varphi_i \tag{6}$$

The representation error is then:

$$\Delta x(M) = x - x'(M) = \sum_{i=1}^{N} y_i \varphi_i - \left(\sum_{i=1}^{M} y_i \varphi_i - \sum_{i=M+1}^{N} b_i \varphi_i \right) = \sum_{i=M+1}^{N} (y_i - b_i) \varphi_i \tag{7}$$

We can measure this representation error by the mean-squared magnitude of $\boldsymbol{\Delta x}$. Our goal is to find the basis vectors φ_i and the constants b_j such that the mean squared error is minimized. We can express the mean-squared error as:

$$\varepsilon^2(M) = \sum_{i=M+1}^{N} \varphi_i^T \sum_x \varphi_i = \sum_{i=M+1}^{N} \varphi_i^T \lambda_i \varphi_i = \sum_{i=M+1}^{N} \lambda_i \tag{8}$$

To minimize the mean-squared error λ_i will have to be the smallest eigenvalues of the covariance matrix. Therefore, we choose the eigenvectors φ_i corresponding to the largest eigenvalues λ_i.of the covariance matrix Σ_x .

The objective of the LDA is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible. We seek to obtain a scalar $\boldsymbol{y}$ by projecting the samples $\boldsymbol{x}$ onto a line.

$$y = W^T x \tag{9}$$

In order to find a good projection vector, we need to define a measure of separation between the projections. Fisher's linear discriminant is a classification method that projects high-dimensional data onto a line and performs classification in this one-dimensional space. The projection maximizes the distance between the means of the two classes while minimizing the variance within each class. Equation 10 defines the Fisher criterion, which is maximized over all linear projections, *w*:

$$J(w) = \frac{\left| \tilde{\mu}_1 - \tilde{\mu}_2 \right|^2}{\tilde{s}_1^2 + \tilde{s}_2^2} \tag{10}$$

Where the within-class scatter is:

$$\left(\tilde{s}_1^2 - \tilde{s}_2^2 \right) = \left(y - \tilde{\mu}_1 \right) + \left(y - \tilde{\mu}_2 \right) \; where \; y \in w \tag{11}$$

Therefore, we will be looking for a projection where examples from the same class are projected very close to each other, and at the same time, the projected means are as far apart as possible.

The KNN classifier is trivial. In the method we simply grow the area surrounding the estimation point $\boldsymbol{x}$ until it encloses a total of k data points. Therefore, the density estimate becomes:

$$P(x) \cong \frac{k}{NA} = \frac{k}{N * A * R_k(x)} \tag{12}$$

here, $R_k(x)$ is the distance between the estimation point $\boldsymbol{x}$ and its k^{th} closest neighbor, and $\boldsymbol{A}$ is simply the area of the unit circle. Classification results follow in figures 2a-2c. These figures show that the eNose data is being classified into three classes.

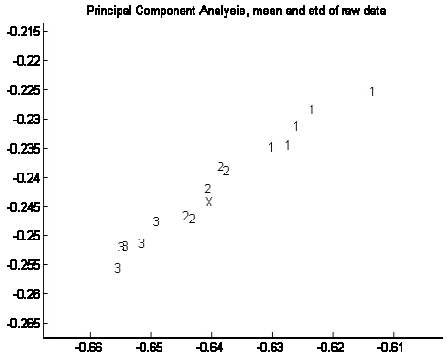

Figure 2a. Principal components classifier

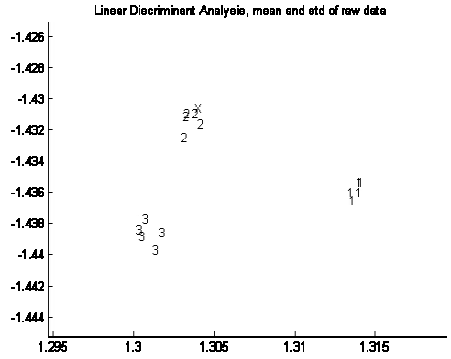

Figure 2bLinear discrimination classifier

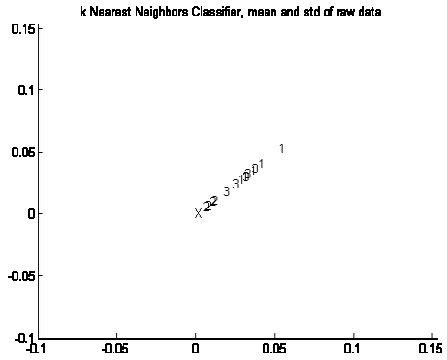

Figure 2c. KNN classifier

All three of the classifiers are able to classify this data correctly. The sample "X" belongs to class 2. The three classifiers, however, work very differently. The PCA is based on signal representation and measures the "closeness" by the mean square error over all points in the distribution. The LDA, on the other hand, maximizes the separation between the means of the projection, while minimizing the variance within each projected class. Now, this fact tells us that the LDA will fail when the discriminatory information is in the variance of the data and not in the mean. The KNN classifier is a very intuitive classifier measuring the "closeness" of the test sample to its k-nearest neighbors.

3 APPLICATION OF PHASE ONLY FILTER ON ENOSE DATA

Previously the performance of the POF was investigated and compared to the KNN classifier by Gudmundsson[3] et.al. The database consists of fifteen samples. Four sensors represent each sample, and two features represent each sensor. Thus the database consists of 120 items. A cross-correlation between the sample and the database was performed to generate the output at the output plane. Each sample can be represented by the following set:

$$sample = \{Sensor1(mean, std), Sensor2(mean, std)Sensor3(mean, std)Sensor4(mean, std)\} \quad (13)$$

and the database by the set of samples:

$$database = \{sample1, ..., sample15\} \quad (14)$$

Therefore, the correlation database simply becomes:

$$corrDB = \{test\_sample \otimes database\} \tag{15}$$

The absolute difference between the test sample and each element in the database provides circular boundary or distance measure between the sample and the database. This fact was used for comparison purposes of the POF and KNN classifier. In the case of the KNN the difference is the direct difference, or distance between the test samples, mean and std, and the mean and std of each element in the database. The distance is simply the computed Euclidean distance, sqrt(mean^2+std^2). In the case of the POF, the auto-correlation of the test sample's mean and std was computed and subtracted from the database of cross-correlations between the sample's mean and std, and the mean and std of each sample in the database. Obviously, a lot of information is lost when the feature space is reduced to two dimensions only. However, the exercise was performed for the sake of comparison between the POF classifier and more conventional classifiers.

If formulated correctly, the POF classifier lends itself much better to a large feature space since the correlation between samples is performed in parallel.

4 THE ONE DIMENSIONAL POF CLASSIFIER

The mathematical foundation of the one dimensional POF filter can easily be derived from simplifying the previously defined two dimensional POF. If we let the Fourier transform of the signal function *f*(x) be denoted by:

$$F(U_x) = \left|F(U_x)\right| \exp(j\Phi U_x) \tag{16}$$

Then a Complex Match Filter (CMF) corresponding to this function *f*(x) at the filter plane is expected to produce the autocorrelation of the input at the output plane. Therefore, the CMF is given by the complex conjugate of the input Fourier spectrum:

$$H_{CMF}(U_x) = F^*(U_x) = \left|F(U_x)\right| \exp(-j\Phi U_x) \tag{17}$$

The inverse Fourier transform of the product of $F(U_x)$ and $H_{CMF}(U_x)$ results in the convolution of *f*(x) and *f*(-x), which is the equivalent of the autocorrelation of *f*(x). Moreover, when $|F(U_x)|$ is set to unity, H_{CMF} becomes a phase only filter:

$$H_{POF}(U_x) = \exp(-j\Phi U_x) \tag{18}$$

Finally, the phase only correlation of the input signal and the reference sample is simply:

$$F^{-1}\{F(U_x) \times H_{POF}(U_x)\} \tag{19}$$

Since the convolution operator in the space domain is equivalent to the product operator in the frequency domain, one can think of the POF as an edge enhancer by way of division by $|F(U_x)|$, and integrator, by integrating the product of the input signal and the reference sample.

5 OPTICAL ARCHITECTURE

For each class a different POF needs to be developed from the database samples. One such approach is to represent the collection of data in 2 dimensions, and perform the comparison in parallel using phase-only filter. For each class, a different POF needs to be developed. A 4f correlator set up may allow for parallel matching of all the input into different classes in parallel. One possible implementation is shown in the following, figure 3.

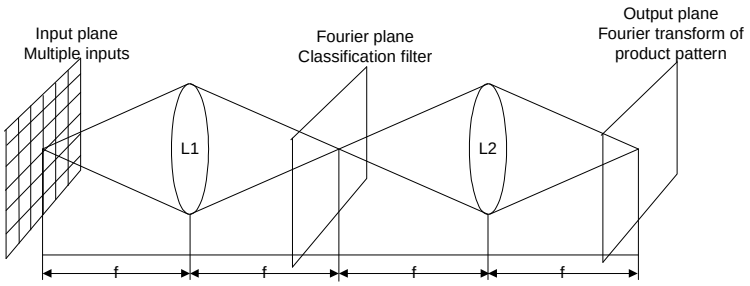

Figure 3. 4 f Cylindrical lens correlator

To attain one dimensional correlation in the optical domain, an array of simple cylindrical lenses L1 are used to perform the Fourier transform. All of the to-be-classified inputs are placed in the input plane, one focal length from the lens, and its Fourier transform forms at the focal distance from the lens in the Fourier plane. The set of classification filters will be at this plane. After the light passes through another array of cylindrical lenses, the Fourier transform of the product pattern will appear at the output plane. This output is expected to be generated as a result of the parallel matching producing different classes at spatially varying locations.

6 DATA REPRESENTATION AND ALGORITHM

The most straightforward approach to represent the multi dimensional data for the optical correlator is by reducing the four dimensional signal to a one dimensional signal by concatenating the output from all four sensors. Therefore, each line on the SLM will consist of eighty points and the database will produce fifteen data lines.

Using cylindrical lenses to process each line from the database simplifies the problem to a manageable size. Each line of the database can be represented by the following set:

$$Line = \{p_1, p_2, ...p_{80}\} \tag{20}$$

Therefore, the entire database can be represented by the following set:

$$db = \{line_1, line_2, ...line_{15}\} \tag{21}$$

Similarly the unknown input sample can be represented by the set:

$$unknown = \{u_1, u_2, ...u_{80}\} \tag{22}$$

To focus on signal features rather than the energy of the signal, we must establish a reference value. Comparing the correlation difference from the auto correlation of the unknown input sample, one can select the best match using the following search algorithm:

$$\forall(lines\,in\,db) \otimes unknown, \exists line \otimes unknown \,|\, line \otimes unknown \approx unknown \otimes unknown \tag{23}$$

This search algorithm truly searches for the best matching signature.

7 EXPERIMENT

An experiment using the eNose data set was designed and implemented in Matlab™ to simulate the optical correlator. The following, figures 4-5, shows the point to point distance between the cross correlation of the data base with the unknown input sample and the autocorrelation of the test sample.

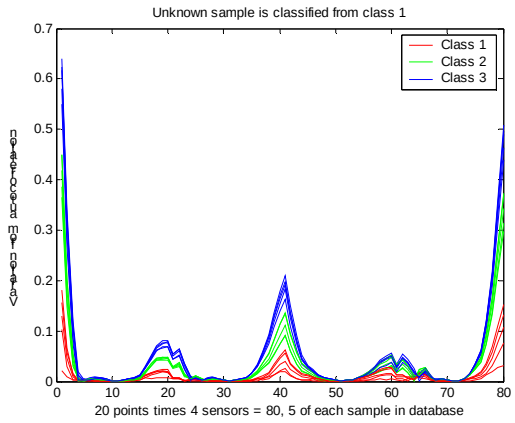

Figure 4. Correlation difference in case of class 1input sample

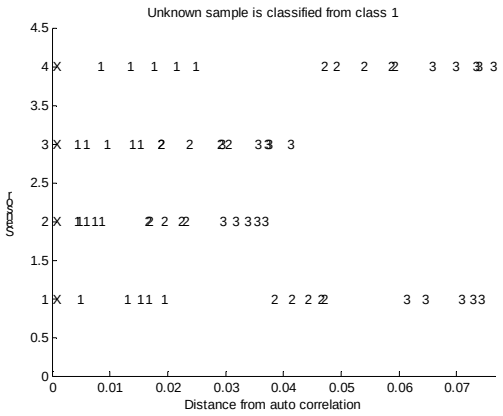

Figure 5. Unknown sample is classified as class 1

Clearly, the different classes from the database group together. All the class 1 samples cluster together, the class 2 samples cluster together, and the class 3 samples all cluster together. The unknown input sample is classified as class 1 type since the distance from the input sample to the database samples is shortest. The average discrimination ratio between the test sample and the classes is: 1.0000 for class 1 to class 1, 3.0939 for the class 1 to class 2, and finally 4.6411 for the Class 1 to the class 3 type.

Similarly, figure 6-7 shows the point to point correlation in the case of the unknown sample being of the class 2 kind. The results are similar. All the class 2 type samples in the database are clustered at the bottom of the graph, and the unknown sample is classified as class 2 type.

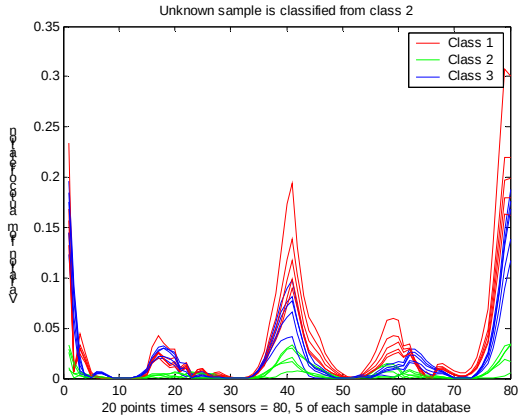

Figure 6. Correlation difference in case of class 2 input sample

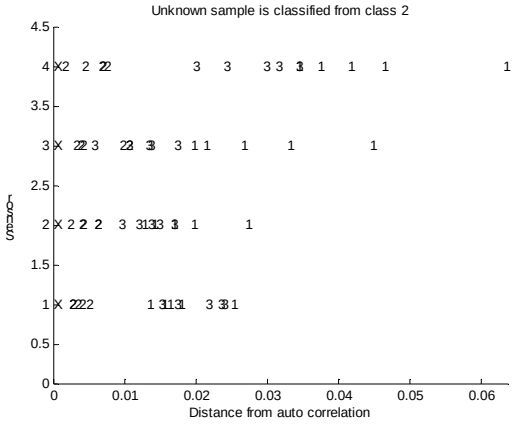

Figure 7. Unknown sample is classified as class 2

The average discrimination ratio between the test sample and the brands in the database is still very high. Between the class 2 and the class 1 the ratio is 5.9072. And between the class 2 and class 3 the discrimination ratio is 3.9475
Finally, figures 8-9 shows the point to point correlations of the case where the unknown sample is of the class 3 type. The results are again very similar. The classes are all well defined and the discrimination ratio is high. Similarly, the class 3 samples are all clustered at the bottom of the graph and the unknown sample is classified as class 3 type.

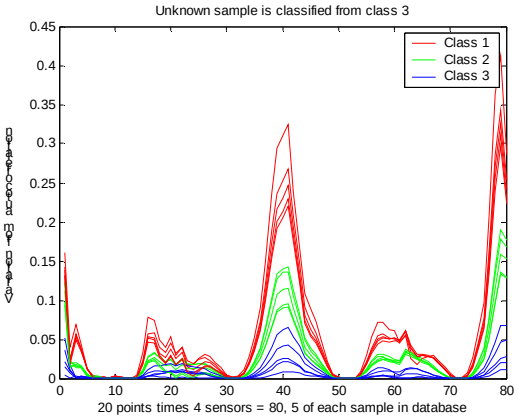

Figure 8. Correlation difference in case of class 3 input sample

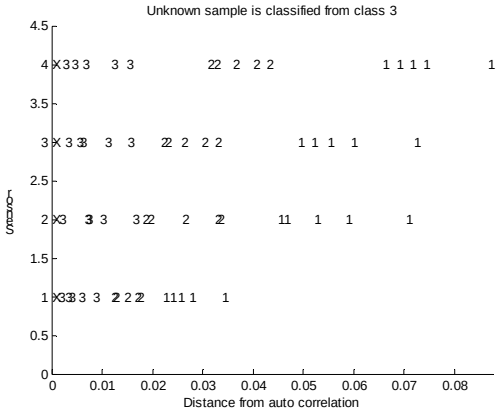

Figure 9. Unknown sample is classified as class 3

As expected, the average discrimination ratio between the test sample and the classes in the database is still very high. Between the class 3 and class 1 type the ratio is 7.6295, while it is 3.7226 between the class 3 and class 2 type. The discrimination ratio can be tabularized for comparison with the previous classification exercise using the mean and standard deviation. In four out of six cases this new method on raw data space outperforms the KNN classification method using mean and std.

Summary of discrimination ratios			
	class 1	**class 2**	**class 3**
POF raw data class 1		3.0939	4.6411
POF mean, STD class 1		**8.7176**	**12.0534**
KNN mean, STD class 1		3.3630	5.4325
POF raw data class 2	**5.9072**		**3.9475**
POF mean, STD class 2	1.1593		1.6197
KNN mean, STD class 2	4.8187		3.6372
POF raw data class 3	**7.6295**	3.7226	
POF mean, STD class 3	4.9150	**4.1119**	
KNN mean, STD class 3	6.3922	3.4211	

Table 1. Comparison of discrimination ratios

However, within a class, the data may exhibit a certain degree of variability. A training method must be developed in order to incorporate the variability into the filter synthesis. The next section describes one possible way to achieve this.

8 CONCLUSIONS

The paper proposes a new problem domain for the POF, namely data from multiple one dimensional sensors. Data from an electronic nose device is used for demonstration purposes. We compare the POF classifier performance on multi dimensional data to the performance of the k-nearest neighbors (KNN) classifier. To take advantage of the POF a suitable multi dimensional data representation for multi dimensional signal is introduced. The one dimensional POF is proposed; and an algorithm is developed, and simulated.

The POF applied to raw data shows very promising results and outperforms the classical KNN classifier for all three classes.

Future work in this area would include the investigation of performance in larger feature space, and implementation of the proposed electro/optical correlator to perform classification on-line.

ACKNOWLEDGEMENT

This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore Laboratory under contract No. W-7405-Eng-48. K. S. Gudmundsson was supported by a grant from Wright State University College of Engineering, while the work was done, and A. A. S. Awwal was with the Lawrence Livermore Laboratories.

REFERENCES

1. H. T. Nagle, S. S. Schiffman and R. Gutierrez-Osuna , "The how and why of electronic noses" *IEEE Spectrum*, September 1998, pp. 22-34.
2. Bellman, R. (1961), Adaptive Control Processes: A Guided Tour, Princeton University Press
3. K. Gudmundsson, A. A. S. Awwal, R. Gutierrez-Osuna, "Application of POF on eNose Data", in *Photonic Devices and Algorithms for Computing V*, K. M. Iftekharuddin and A. A. S. Awwal, Editors, Proceedings of SPIE 5201, pp. 117-125, 2003.